```
 1  HIMEM: 32767
 2  HOME : PRINT "FROB MONITOR PROGRAM"
 3  PRINT : REM  VERSION 1.3 1/24/83
 4  DIM DIG%(4),S%(32),BP%(3),PG%(18),BIT%(8)
 5 FM% = 0
 6  DIM OCODE%(255),ACODE%(255)
 7 PM% = 768:BUF$ = "8":BUF = 8 * 4096
 8  PRINT  CHR$ (4);"BLOAD PMOVE.OBJ"
 10  INPUT "WHAT IS THE FROB SLOT NUMBER (1,7)?";FSLOT%
 12  IF FSLOT% < 1 OR FSLOT% > 7 THEN 10
 14 DEV = 4096 * 12 + 16 * (FSLOT% + 8)
 20  INPUT "DO YOU WISH TO LOAD A FILE? (Y/N)";A$
 22  IF A$ = "N" THEN 30
 24  IF A$ <  > "Y" THEN 20
 26  GOSUB 11000: REM  GO LOAD A FILE
 30  INPUT "DO YOU WISH TO USE FMON? (Y/N)";A$
 32  IF A$ = "N" THEN 40
 34  IF A$ <  > "Y" THEN 30
 36  GOSUB 12000: REM  GO LOAD FMON
 40 BP% = 0: REM    SET TO 1 ON BREAK POINT HIT
 60  GOSUB 4000: REM  LOAD THE SHADOW
 100  IF FM% = 0 THEN 200
 110  INPUT "PLEASE TURN ON VCS AND HIT RETURN";A$
 130  IF  PEEK (DEV) < 128 THEN 110
 140  POKE DEV + 1,0
 150  IF  PEEK (DEV) < 128 THEN 110
 160  PRINT "SETTING UP...";
 200 OPCODES$ = "***ADCANDASLBCCBCSBEQBITBMIBNEBPLBRKBVCBVSCLCCLDCLICLVCMPCPXCPY"
 201 OPCODES$ = OPCODES$ + "DECDEXDEYEORINCINXINYJMPJSRLDALDXLDYLSRNOPORAPHAPHPPLA"
 202 OPCODES$ = OPCODES$ + "PLPROLRORRTIRTSSBCSECSEDSEISTASTXSTYTAXTAYTSXTXATXSTYA"
 210  FOR I = 0 TO 255: READ OCODE%(I): NEXT I
 211  DATA  11,35,0,0,0,35,3,0
 212  DATA  37,35,3,0,0,35,3,0
 213  DATA  10,35,0,0,0,35,3,0
 214  DATA  14,35,0,0,0,35,3,0
 215  DATA  29,2,0,0,7,2,40,0
 216  DATA  39,2,40,0,7,2,40,0
 217  DATA  8,2,0,0,0,2,40,0
 218  DATA  45,2,0,0,0,2,40,0
 219  DATA  42,24,0,0,0,24,33,0
 220  DATA  36,24,33,0,28,24,33,0
 221  DATA  12,24,0,0,0,24,33,0
 222  DATA  16,24,0,0,0,24,33,0
 223  DATA  43,1,0,0,0,1,41,0
 224  DATA  38,1,41,0,28,1,41,0
 225  DATA  13,1,0,0,0,1,41,0
 226  DATA  47,1,0,0,0,1,41,0
 227  DATA  0,48,0,0,50,48,49,0
 228  DATA  23,0,54,0,50,48,49,0
 229  DATA  4,48,0,0,50,48,49,0
 230  DATA  56,48,55,0,0,48,0,0
 231  DATA  32,30,31,0,32,30,31,0
 232  DATA  52,30,51,0,32,30,31,0
 233  DATA  5,30,0,0,32,30,31,0
 234  DATA  17,30,53,0,32,30,31,0
 235  DATA  20,18,0,0,20,18,21,0
 236  DATA  27,18,22,0,20,18,21,0
 237  DATA  9,18,0,0,0,18,21,0
 238  DATA  15,18,0,0,0,18,21,0
 239  DATA  19,44,0,0,19,44,25,0
 240  DATA  26,44,34,0,19,44,25,0
 241  DATA  6,44,0,0,0,44,25,0
 242  DATA  46,44,0,0,0,44,25,0
 250  FOR I = 0 TO 255: READ ACODE%(I): NEXT I
 260  DATA  1,10,1,1,1,3,3,1
 261  DATA  1,5,2,1,1,4,4,1
 262  DATA  13,11,1,1,1,6,6,1
 263  DATA  1,9,1,1,1,8,8,1
 264  DATA  4,10,1,1,3,3,3,1
 265  DATA  1,5,2,1,4,4,4,1
 266  DATA  13,11,1,1,1,6,6,1
 267  DATA  1,9,1,1,1,8,8,1
 268  DATA  1,10,1,1,1,3,3,1
 269  DATA  1,5,2,1,4,4,4,1
 270  DATA  13,11,1,1,1,6,6,1
 271  DATA  1,9,1,1,1,8,8,1
 272  DATA  1,10,1,1,1,3,3,1
 273  DATA  1,5,2,1,12,4,4,1
 274  DATA  13,11,1,1,1,6,6,1
 275  DATA  1,9,1,1,1,8,8,1
 276  DATA  1,10,1,1,3,3,3,1
 277  DATA  1,1,1,1,4,4,4,1
 278  DATA  13,11,1,1,6,6,7,1
 279  DATA  1,9,1,1,1,8,1,1
 280  DATA  5,10,5,1,3,3,3,1
 281  DATA  1,5,1,1,4,4,4,1
 282  DATA  13,11,1,1,6,6,7,1
```

```
 283  DATA  1,9,1,1,8,8,9,1
 284  DATA  5,10,1,1,3,3,3,1
 285  DATA  1,5,1,1,4,4,4,1
 286  DATA  13,11,1,1,1,6,6,1
 287  DATA  1,9,1,1,1,8,8,1
 288  DATA  5,10,1,1,3,3,3,1
 289  DATA  1,5,1,1,4,4,4,1
 290  DATA  13,11,1,1,1,6,6,1
 291  DATA  1,9,1,1,1,8,8,1
 390  REM  HERE IS THE SLEEP ROUTINE
 391  REM  TO BE RUN IN PAGE 0 OF VCS
 400  FOR I = 1 TO 18: READ PG%(I): NEXT I
 401  DATA   198,241,208,252
 402  DATA   198,240,208,248
 403  DATA   198,239,208,244
 404  DATA   108,250,255,2,0,0
 405  REM  E0: DEC $F1
 406  REM  E2: BNE $E0
 407  REM  E4: DEC $F0
 408  REM  E6: BNE $E0
 409  REM  E8: DEC $EF
 410  REM  EA: BNE $E0
 411  REM   EC: JMP ($FFFA)
 412  REM  EF: DFB 2,0,0
 420  PRINT "READY"
 490 PC = 1000
 495 SF% = 0
 500 FAIL% = 0
 510  GOSUB 1000
 515  IF A$ = "RESTORE" THEN 800
 517  IF  LEN (D$) = 0 THEN D$ = "1"
 520  IF  LEN (B$) > 4 THEN 900
 540  IF  LEN (C$) > 0 THEN 600
 550  IF  LEN (B$) = 0 THEN 500
 570  GOTO 20000: REM  THE READ MEM ROUTINE
 600  REM  DELIM FIELD NOT BLANK
 610  IF C$ = "." THEN 20000: REM  READ MEMORY
 620  IF C$ = "<" THEN 21000: REM  WRITE MEMORY
 630  IF C$ = "M" THEN 22000: REM  MODIFY REGISTER
 640  IF C$ = "R" THEN 23000: REM  DISPLAY REGISTERS
 650  IF C$ = "P" THEN 24000: REM  PLACE BREAKPOINT
 660  IF C$ = "K" THEN 25000: REM  KILL BREAK POINT
 670  IF C$ = "H" THEN 26000: REM  HALT THE VCS
 675  IF C$ = "S" THEN 31000: REM  START VCS AT RESET
 680  IF C$ = "G" AND  LEN (B$) = 0 THEN 27000: REM   GO FROM BREAK
 690  IF C$ = "G" THEN 28000: REM  GOTO ADDR
 700  IF C$ = "L" AND  LEN (B$) = 0 THEN 29000: REM  CONTINUE DISASM
 710  IF C$ = "L" THEN 30000: REM  DISASM
 720  IF C$ = "I" THEN 32000: REM  LOAD FILE
 750  GOTO 950
 760  GOTO 500
 800  REM  TURN ON THE I/O PORTS
 810  POKE DEV,48
 820  PRINT "COMMUNICATION IS NOW ON"
 830  IF FM% = 0 THEN 500
 840  PRINT "TURN THE VCS OFF AND ON TO GET TO FMON"
 850  GOTO 500
 900  PRINT "ERROR"
 910  GOTO 500
 920  PRINT "ERROR -- FMON NOT LOADED"
 930  GOTO 500
 950  PRINT "NOT A VALID COMMAND"
 960  GOTO 500
1000  REM  ROUTINE TO RETURN STRING
1005 B$ = "":C$ = "":D$ = ""
1010  INPUT ">";A$
1015 K% = 1
1020  FOR L = 1 TO  LEN (A$)
1025 X$ =  MID$ (A$,L,1)
1027  IF X$ = " " THEN 1100
1030  GOSUB 2000: REM  IS THIS A HEX DIG?
1031  IF FLAG% > 0 AND K% = 2 THEN K% = 3
1032  IF FLAG% = 0 THEN K% = K% + 1
1033  IF K% > 3 THEN 1100
1035  IF K% = 1 THEN B$ = B$ + X$
1040  IF K% = 2 THEN C$ = C$ + X$
1050  IF K% = 3 THEN D$ = D$ + X$
1100  NEXT L
1110  RETURN
2000  REM  RETURN FLAG% SET IF X$ IS HEX
2005 FLAG% = 1
2010  IF X$ < "0" OR X$ > "F" THEN FLAG% = 0
2020  IF X$ > "9" AND X$ < "A" THEN FLAG% = 0
2100  RETURN
2500  REM   CHANGE B$ TO HEX ADDR
2505 ADDR = 0
2510  FOR L = 1 TO  LEN (B$)
```

```
2520 X% =  ASC ( MID$ (B$,L,1)) - 48
2540  IF X% > 9 THEN X% = X% - 7
2550 ADDR = ADDR * 16 + X%
2560  NEXT L
2570  RETURN
2630  GOTO 500
3000  REM  ROUTINE TO TALK TO FMON
3100  IF  PEEK (DEV) < 128 THEN 3100
3110  POKE DEV + 1,WR%
3120  RETURN
3200  REM  READ VALUE FROM FROB
3202 X% =  PEEK (DEV): IF X% > 127 THEN X% = X% - 128
3205  IF X% < 64 THEN 3202
3210 X% =  PEEK (DEV + 1)
3220  RETURN
3300  REM  ROUTINE TO GET REGISTERS
3310  GOSUB 3200: POKE BUF + 4086,X%
3311  GOSUB 3200: POKE BUF + 4085,X%
3312  GOSUB 3200: POKE BUF + 4087,X%
3313  GOSUB 3200: POKE BUF + 4088,X%
3314  GOSUB 3200: POKE BUF + 4089,X%
3320  FOR I = 1 TO 32
3330  GOSUB 3200:S%(I) = X%
3340  NEXT I
3345  GOSUB 3200
3350 N% = 0: IF  PEEK (BUF + 4085) = 128 THEN N% = 1
3360 Z% = 1: IF N% = 1 THEN Z% = 0
3370  IF  PEEK (BUF + 4085) = 1 THEN Z% = 0
3380  GOSUB 13000: REM  TAKE FLAGS APART
3385  POKE BUF + 4084,N% * 128 + BIT%(2) * 64 + BIT%(3) * 32 + BIT%(4) * 16 + BIT%(5) * 8 + BIT%(6) * 4 + Z% * 2 + BIT%(8)
3390  RETURN
3400  REM  ROUTINE TO RESTORE REGISTERS
3410  FOR I = 32 TO 1 STEP  - 1
3420 WR% = S%(I): GOSUB 3000
3430  NEXT I
3440  RETURN
4000  REM  LOAD SHADOW TO FROB
4010  POKE PM% + 18,FSLOT%
4020  POKE PM% + 19, INT (BUF / 256)
4030  POKE PM% + 20,16
4040  POKE PM% + 21,32
4050  CALL PM% + 22
4060  POKE DEV,48
4100  RETURN
5000  REM  CONVERT D$ TO LENGTH FIELD
5005  IF D$ = "1000" THEN 5200
5010 NUMBYT% = 0
5020  IF  LEN (D$) = 0 THEN  RETURN
5030  IF  LEN (D$) > 3 THEN D$ =  RIGHT$ (D$,3)
5040  FOR L = 1 TO  LEN (D$)
5050 X% =  ASC ( MID$ (D$,L,1)) - 48
5060  IF X% > 9 THEN X% = X% - 7
5070 NUMBYT% = NUMBYT% * 16 + X%
5080  NEXT L
5100  RETURN
5200 NUMBYT% = 4096: RETURN
6000  REM    PRINT LOWER 3 HEX DIGITS OF A1
6010 A2 =  INT (A1 / 256):A3 = A1 - (A2 * 256)
6020  POKE 912,A2
6030  CALL 925: REM  PRINT HEX DIGIT
6040  POKE 912,A3
6050  CALL 919: REM  PRINT HEX BYTE
6100  RETURN
6200  RETURN
6500  REM   PRINT 2 HEX DIGITS FROM A1
6510  POKE 912,A1
6520  CALL 919: REM  PRINT HEX DIGITS
6530  RETURN
7000  REM  FMON READ
7010 WR% = 16: GOSUB 3000
7020 WR% = ADDR / 256: GOSUB 3000
7030 WR% = ADDR - (WR% * 256): GOSUB 3000
7040 WR% = 1: GOSUB 3000
7050  GOSUB 3200
7060 A1 = X%
7070  RETURN
8000  REM  DO AN FMON WRITE
8010 WR% = 32: GOSUB 3000
8020 WR% = ADDR / 256: GOSUB 3000
8030 WR% = ADDR - (WR% * 256): GOSUB 3000
8040 WR% = 1: GOSUB 3000
8050 WR% = A1: GOSUB 3000
8100  RETURN
9000  REM  ROUTINE TO PUT TO SLEEP
9010 WR% = 32: GOSUB 3000
9020 WR% = 0: GOSUB 3000
9030 WR% = 224: GOSUB 3000
```

```
 9040 WR% = 18: GOSUB 3000
 9045  FOR I = 1 TO 18
 9050 WR% = PG%(I): GOSUB 3000
 9060  NEXT I
 9062 WR% = 64: GOSUB 3000
 9064 WR% = 0: GOSUB 3000
 9066 WR% = 224: GOSUB 3000
 9080  GOSUB 4000: REM  LOAD SHADOW
 9100  RETURN
 11000  REM   LOAD A FILE
 11010  INPUT "WHAT IS THE FILE NAME?";A$
 11020  INPUT "LOAD TO WHAT VCS HEX ADDRESS?";B$
 11030  IF  LEN (B$) <  > 4 THEN 11020
 11031 X$ =  LEFT$ (B$,1)
 11032  IF X$ = "2" OR X$ = "4" OR X$ = "6" OR X$ = "0" OR X$ = "8" OR X$ = "A" OR X$ = "C" OR X$ = "E" THEN 11020
 11035 OK% = 1
 11040  FOR L = 1 TO 4
 11050 X$ =  MID$ (B$,L,1)
 11060  GOSUB 2000: REM  SEE IF A HEX DIGIT
 11070  IF FLAG% = 0 THEN OK% = 0
 11080  NEXT L
 11090  IF OK% = 0 THEN 11020
 11100  PRINT  CHR$ (4);"BLOAD ";A$;",A$";BUF$; RIGHT$ (B$,3)
 11110  RETURN
 12000  REM : LOAD FMON
 12010  PRINT  CHR$ (4);"BLOAD FMON.OBJ,A$";BUF$;"F00"
 12020 FM% = 1
 12030  RETURN
 13000  REM  CONVERT X% TO BITS
 13005 X1% = X%:XB% = 256
 13010  FOR I = 1 TO 8
 13020 BIT%(I) = 0:XB% = XB% / 2
 13030  IF X1% < XB% THEN 13060
 13040 BIT%(I) = 1
 13050 X1% = X1% - XB%
 13060  NEXT I
 13070  RETURN
 14000  REM  PRINT DADDR AND ":"
 14010 X = DADDR: GOSUB 14300
 14040  PRINT ":";
 14050  RETURN
 14100  REM  PRINT 2 HEX FROM X%
 14110 A1 = X%
 14120  GOTO 6500
 14200  REM  PRINT OPCODE
 14210 N% = OCODE%(OP%) * 3 + 1
 14220  PRINT  MID$ (OPCODES$,N%,3);
 14230  RETURN
 14300  REM  PRINT DADDR RELATIVE
 14310  PRINT FHR$;
 14320 A1 = X: GOSUB 6000
 14330  RETURN
 20000  REM  READ MEMORY
 20002 RD% = 1: REM  FOLLOW THE READ PATH
 20005  GOSUB 5000: REM  CONVERT D$ TO NUMBER
 20010  IF  LEN (B$) = 4 THEN 20050
 20020  FOR L = 1 TO 4 -  LEN (B$)
 20030 B$ = "0" + B$
 20040  NEXT L
 20050 X$ =  LEFT$ (B$,1)
 20055 FHR$ = X$
 20060  IF X$ = "0" OR X$ = "2" OR X$ = "4" OR X$ = "6" OR X$ = "8" OR X$ = "A" OR X$ = "C" OR X$ = "E" THEN 20100
 20070 B$ = BUF$ +  RIGHT$ (B$,3)
 20080  GOTO 20200
 20100 B$ = "0" +  RIGHT$ (B$,3)
 20110  IF FM% = 0 THEN 920
 20200  GOSUB 2500: REM  CONVERT ADDRESS
 20201  IF RD% = 0 THEN 21100: REM  DO WRITE
 20203  PRINT
 20205  PRINT FHR$;
 20210 A1 = ADDR: GOSUB 6000
 20212  PRINT ":";
 20220  IF ADDR > 4095 THEN A1 =  PEEK (ADDR)
 20225  IF ADDR < 256 THEN 20600
 20230  IF ADDR < 4096 THEN  GOSUB 7000: REM     GO DO FMON READ
 20240  GOSUB 6500
 20250  PRINT " ";
 20260 ADDR = ADDR + 1:NUMBYT% = NUMBYT% - 1
 20270  IF NUMBYT% = 0 THEN 20500
 20280 B% = ADDR - ( INT (ADDR / 8) * 8)
 20290  IF B% = 0 THEN 20203
 20300  GOTO 20220
 20500  PRINT : PRINT
 20510  GOTO 500
 20600  IF ADDR < 224 THEN 20230
 20610 A1 = S%(ADDR - 223)
 20620  GOTO 20240
```

```
 21000  REM  WRITE MEMORY
 21010 RD% = 0: REM  CHOOSE WRITE PATH
 21020  GOTO 20005
 21100  IF NUMBYT% > 255 THEN 900
 21110 A1 = NUMBYT%
 21120  IF ADDR > 4095 THEN  POKE ADDR,A1
 21125  IF ADDR < 256 THEN 21200
 21130  IF ADDR < 4096 THEN  GOSUB 8000
 21140 NUMBYT% = 1
 21150  GOTO 20203
 21200  IF ADDR < 224 THEN 21130
 21210 S%(ADDR - 223) = A1
 21220  GOTO 21140
 22000  REM  MODIFY REGISTER
 22005  IF FM% = 0 THEN 920
 22010  INPUT "WHICH REGISTER? (A,X,Y,S,P)";A$
 22020  IF A$ <  > "A" AND A$ <  > "X" AND A$ <  > "Y" AND A$ <  > "S" AND A$ <  > "P" THEN 22010
 22030  INPUT "NEW VALUE (HEX)=";D$
 22040  IF  LEN (D$) > 2 THEN 22030
 22050 A1 = 0:FLAG% = 1
 22060  FOR I = 1 TO  LEN (D$)
 22065  IF FLAG% = 0 THEN 22100
 22070 X$ =  MID$ (D$,I,1): GOSUB 2000
 22080 X% =  ASC (X$) - 48: IF X% > 9 THEN X% = X% - 7
 22090 A1 = A1 * 16 + X%
 22100  NEXT I
 22110  IF FLAG% = 0 THEN 22030
 22120  IF A$ <  > "A" THEN 22130

 22125  POKE BUF + 4086,A1: GOTO 500
 22130  IF A$ <  > "X" THEN 22140
 22135  POKE BUF + 4087,A1: GOTO 500
 22140  IF A$ <  > "Y" THEN 22150
 22145  POKE BUF + 4088,A1: GOTO 500
 22150  IF A$ <  > "S" THEN 22160
 22152 SF% = 1: REM  SHOW THE STACK POINTER AS UPDATED
 22155  POKE BUF + 4089,A1: GOTO 500
 22160 F% = A1:F% = F% / 2
 22165 Z% = F% - ( INT (F% / 2) * 2)
 22170 N% =  INT (F% / 64)
 22175  IF Z% = 0 OR N% = 0 THEN 22200
 22180  PRINT "??? EITHER Z OR N OR BOTH MUST BE 0"
 22190  GOTO 22030
 22200  POKE BUF + 4084,A1
 22210  IF Z% = 0 THEN 22250
 22220  POKE BUF + 4085,255
 22230  GOTO 500
 22250  IF N% THEN 22280
 22260  POKE BUF + 4085,1
 22270  GOTO 500
 22280  POKE BUF + 4085,128
 22290  GOTO 500
 23000  REM  DISPLAY REGISTERS
 23005  IF FM% = 0 THEN 920
 23010  PRINT
 23020  PRINT "A=";:A1 =  PEEK (BUF + 4086): GOSUB 6500
 23030  PRINT " X=";:A1 =  PEEK (BUF + 4087): GOSUB 6500
 23040  PRINT " Y=";:A1 =  PEEK (BUF + 4088): GOSUB 6500
 23050  PRINT " S=";:A1 =  PEEK (BUF + 4089): GOSUB 6500
 23060  PRINT " P=";:A1 =  PEEK (BUF + 4084): GOSUB 6500
 23070  PRINT : GOTO 500
 24000  REM  PLACE BREAK POINT
 24005  IF FM% = 0 THEN 920
 24010  IF  LEN (B$) <  > 4 THEN 950
 24020 B$ = BUF$ +  RIGHT$ (B$,3)
 24030  GOSUB 2500
 24040 BREAK = ADDR
 24050  FOR I = 1 TO 3
 24060 BP%(I) =  PEEK (ADDR + I - 1)
 24070  NEXT I
 24075  REM  STICK IN JUMP TO FF00
 24080  POKE ADDR,76: POKE ADDR + 1,00: POKE ADDR + 2,255
 24090  GOTO 500
 25000  REM  KILL BREAK POINT
 25005  IF A$ <  > "K" THEN 950
 25010  IF BREAK = 0 THEN 500
 25020  FOR I = 1 TO 3
 25030  POKE BREAK + I - 1,BP%(I)
 25040  NEXT I
 25050 BREAK = 0: GOTO 500
 26000  REM  HALT THE VCS
 26005  IF A$ <  > "H" THEN 950
 26010  POKE DEV,0
 26020  PRINT "FROB MEMORY NOW UNDER APPLE CONTROL"
 26030  GOTO 500
 27000  REM   GO FROM BREAK
 27002  IF FM% = 0 THEN 920
```

```
 27003  IF PC = 1000 THEN 27200
 27005  IF PC = BREAK - BUF + 61440 THEN 27100
 27006  INPUT "DO YOU WISH TO WAIT FOR A BREAK? (Y/N)";A$
 27007  IF A$ <  > "Y" AND A$ <  > "N" THEN 27006
 27010  POKE BUF + 3990,PC - ( INT (PC / 256) * 256)
 27015  POKE BUF + 3991, INT (PC / 256)
 27020  GOSUB 9000: REM  SLEEP AND SET UP
 27030 WR% = 48: GOSUB 3000
 27040  GOSUB 3400: REM  RESTORE REGISTERS
 27042  IF A$ = "N" THEN 500
 27045  PRINT "WAITING FOR BREAK POINT"
 27050  GOSUB 3300: REM   WAIT FOR BREAK
 27052  PRINT "BREAK POINT ENCOUNTERED"
 27053 SF% = 1: REM  MARK STACK POINTER AS UPDATED
 27055 PC = BREAK - BUF + 61440
 27060  GOTO 25010
 27100  PRINT "ERROR -- YOU MUST MOVE THE BREAKPOINT"
 27110  PRINT "BEFORE YOU CONTINUE"
 27120  GOTO 500
 27200  PRINT "ERROR -- NO BREAK POINT FROM"
 27210  PRINT "WHICH TO PROCEED"
 27220  GOTO 500
 28000  REM  GOTO ADDR
 28005  IF SF% > 0 THEN 28010
 28007  PRINT "WARNING -- STACK POINTER SET AT ";
 28008 A1 =  PEEK (BUF + 4089): GOSUB 6500: PRINT
 28010  GOSUB 2500
 28020 PC = ADDR
 28030  GOTO 27006
 29000  REM  CONTINUE DISASM
 29005 DLIM = DADDR + 30
 29010 OP% =  PEEK (DADDR)
 29015  GOSUB 14000: REM   PRINT DADDR:
 29016 DADDR = DADDR + 1
 29017 X% = OP%: GOSUB 14100: PRINT " ";
 29020  ON ACODE%(OP%) GOTO 29100,29150,29200,29250,29300,29350,29400,29450,29500,29550,29600,29650,29700
 29100  REM  IMPLIED (NO ADDRESS)
 29110  PRINT "      ";
 29120  GOSUB 14200: REM  PRINT OP STRING
 29130  GOTO 29900
 29150  REM  ACCUMULATOR
 29155  PRINT "      ";
 29160  GOSUB 14200
 29165  PRINT " A";
 29170  GOTO 29900
 29200  REM  PAGE 0 ADDRESS
 29205 X1% =  PEEK (DADDR):DADDR = DADDR + 1
 29210 X% = X1%: GOSUB 14100: PRINT " ";
 29215  PRINT "   ";
 29220  GOSUB 14200: PRINT " $";
 29225 X% = X1%: GOSUB 14100
 29230  GOTO 29900
 29250  REM  16 BIT ABSOLUTE
 29252 X1% =  PEEK (DADDR):DADDR = DADDR + 1
 29254 X% = X1%: GOSUB 14100: PRINT " ";
 29256 X2% =  PEEK (DADDR):DADDR = DADDR + 1
 29258 X% = X2%: GOSUB 14100: PRINT " ";
 29260  GOSUB 14200: PRINT " $";
 29262 X% = X2%: GOSUB 14100
 29264 X% = X1%: GOSUB 14100
 29270  GOTO 29900
 29300  REM  IMMEDIATE

 29305 X1% =  PEEK (DADDR):DADDR = DADDR + 1
 29310 X% = X1%: GOSUB 14100: PRINT " ";
 29315  PRINT "   ";
 29320  GOSUB 14200: PRINT " #$";
 29325 X% = X1%: GOSUB 14100
 29330  GOTO 29900
 29350  REM  ZERO PAGE INDEXED X
 29355 X1% =  PEEK (DADDR):DADDR = DADDR + 1
 29360 X% = X1%: GOSUB 14100: PRINT "    ";
 29365  GOSUB 14200: PRINT " $";
 29370 X% = X1%: GOSUB 14100: PRINT ",X";
 29375  GOTO 29900
 29400  REM  ZERO PAGE INDEXED Y
 29405 X1% =  PEEK (DADDR):DADDR = DADDR + 1
 29410 X% = X1%: GOSUB 14100: PRINT "    ";
 29415  GOSUB 14200: PRINT " $";
 29420 X% = X1%: GOSUB 14100: PRINT ",Y";
 29425  GOTO 29900
 29450  REM  ABSOLUTE INDEXED X
 29455 X1% =  PEEK (DADDR):DADDR = DADDR + 1
 29460 X% = X1%: GOSUB 14100: PRINT " ";
 29465 X2% =  PEEK (DADDR):DADDR = DADDR + 1
 29470 X% = X2%: GOSUB 14100: PRINT " ";
 29475  GOSUB 14200: PRINT " $";
```

```
29480 X% = X2%: GOSUB 14100
29485 X% = X1%: GOSUB 14100
29490  PRINT ",X";: GOTO 29900
29500  REM  ABSOLUTE INDEXED Y
29505 X1% =  PEEK (DADDR):DADDR = DADDR + 1
29510 X% = X1%: GOSUB 14100: PRINT " ";
29515 X2% =  PEEK (DADDR):DADDR = DADDR + 1
29520 X% = X2%: GOSUB 14100: PRINT " ";
29525  GOSUB 14200: PRINT " $";
29530 X% = X2%: GOSUB 14100
29535 X% = X1%: GOSUB 14100
29540  PRINT ",Y";: GOTO 29900
29550  REM  INDEXED INDIRECT X
29555 X1% =  PEEK (DADDR):DADDR = DADDR + 1
29560 X% = X1%: GOSUB 14100: PRINT "    ";
29565  GOSUB 14200: PRINT " ($";
29570 X% = X1%: GOSUB 14100: PRINT ",X)";
29575  GOTO 29900
29600  REM  INDIRECT INDEXED Y
29605 X1% =  PEEK (DADDR):DADDR = DADDR + 1
29610 X% = X1%: GOSUB 14100: PRINT "    ";
29615  GOSUB 14200: PRINT " ($";
29620 X% = X1%: GOSUB 14100: PRINT "),Y";
29625  GOTO 29900
29650  REM  JUMP INDIRECT
29655 X1% =  PEEK (DADDR):DADDR = DADDR + 1
29660 X% = X1%: GOSUB 14100: PRINT " ";
29661 X2% =  PEEK (DADDR):DADDR = DADDR + 1
29662 X% = X2%: GOSUB 14100: PRINT " ";
29665  GOSUB 14200: PRINT " ($";
29670 X% = X2%: GOSUB 14100
29671 X% = X1%: GOSUB 14100: PRINT ")";
29675  GOTO 29900
29700  REM  RELATIVE BRANCH
29705 X1% =  PEEK (DADDR):DADDR = DADDR + 1
29710 X% = X1%: GOSUB 14100: PRINT "    ";
29715  GOSUB 14200: PRINT " $";
29720  IF X1% < 128 THEN 29750
29730 X1% =  - 1 * (256 - X1%)
29750 X = DADDR + X1%
29760  GOSUB 14300: REM  PRINT 16 BITS SPECIAL FIELD
29900  PRINT
29910  IF DADDR < DLIM THEN 29010
29920  GOTO 500
30000  REM  DISASSEMBLE
30010  IF  LEN (B$) = 4 THEN 30020
30012  PRINT "ERROR -- ADDRESS MUST POINT TO ROM"
30015  GOTO 500
30020  IF  LEN (D$) = 0 THEN D$ = "1"
30025  GOSUB 5000: REM  CONVERT D$
30030 X$ =  LEFT$ (B$,1)
30040  IF X$ = "0" OR X$ = "2" OR X$ = "4" OR X$ = "6" OR X$ = "8" OR X$ = "A" OR X$ = "C" OR X$ = "E" THEN 30012
30050 FHR$ = X$
30060 B$ = BUF$ +  RIGHT$ (B$,3)
30070  GOSUB 2500:DADDR = ADDR
30080 DLIM = DADDR + NUMBYT%
30090  GOTO 29010
31000  REM  START THE VCS
31003  IF A$ <  > "S" THEN 950
31005  GOSUB 4000: REM  RELOAD SHADOW
31010  POKE DEV,16
31020  PRINT "FROB MEMORY NOW UNDER VCS CONTROL"
31025  IF FM% = 0 THEN 31030
31027  PRINT "(COMMUNICATION WITH FMON IS TURNED OFF)"
31030  PRINT "PLEASE POWER UP THE VCS TO START IT"
31040  GOTO 500
32000  REM  LOAD A FILE
32010  IF B$ = "F" AND D$ = "LE" THEN 32020
32015  GOTO 750
32020  GOSUB 11000
32030 FM% = 0
32040  INPUT "DO YOU WISH TO USE FMON? (Y/N)";A$
32050  IF A$ = "N" THEN 500
32060  IF A$ <  > "Y" THEN 32030
32070  GOSUB 12000: REM  LOAD FMON
32080 BP% = 0
32090  GOSUB 4000: REM  LOAD SHADOW
32100  INPUT "PLEASE TURN ON THE VCS AND HIT RETURN";A$
32110  GOTO 500
```